Claims

The following is a copy of Applicant's claims that identifies language being added with underlining ("___") and language being deleted with strikethrough ("~~—~~"), as is applicable:

1. (Currently amended) A method for pausing a transfer of data, the method comprising ~~the steps of~~:

establishing at least one threshold value that is below a maximum number of errors beyond which the errors are uncorrectable;

determining a number of errors detected in the data being transferred;

comparing the number of errors with the at least one threshold value; and

pausing the transfer of data when the number of errors exceeds the at least one threshold value.

2. (Currently amended) The method of claim 1, wherein ~~the step of establishing~~ selecting at least one threshold value comprises establishing a full error threshold value that designates a maximum number of allowable full errors in a codeword of the data being transferred and an erasure threshold value that designates a maximum number of allowable erasures in said codeword.

3. (Currently amended) The method of claim 2, wherein ~~the step of~~ determining the number of errors comprises determining the number of full errors and erasures in the codeword.

4. (Currently amended) The method of claim 3, wherein ~~the step of~~ comparing comprises:

comparing the number of full errors with the full error threshold value; and

comparing the number of erasures with the erasure threshold value.

5. (Original) The method of claim 2, wherein the full error threshold value and the erasure threshold value are established at levels below maximum levels at which an ECC decoder is capable of detecting and correcting errors.

6. (Currently amended) A method of claim 1, wherein ~~the step of~~ pausing the transfer of data comprises preserving the state of the data at the time of the pausing.

7. (Currently amended) The method of claim 1, wherein ~~the step of~~ establishing at least one threshold value comprises setting at least one level at which the errors in said codeword are recoverable.

8. (Currently amended) The method of claim 7, further comprising ~~the step of~~ sparing the locations that contain recoverable errors.

9. (Currently amended) A system for pausing a transfer of data, the system comprising:

means for providing a full error threshold value that is below a maximum number of errors beyond which the errors are uncorrectable;

means for determining the number of full errors in data being transferred from a data storage means;

means for comparing said number of full errors with said full error threshold value; and

means for pausing the transfer of data when said number of full errors exceeds said full error threshold value.

10. (Original) The system of claim 9, further comprising means for determining the number of erasures in said data being transferred from said data storage means.

11. (Original) The system of claim 10, further comprising means for comparing said number of erasures with an erasure threshold value.

12. (Original) The system of claim 11, wherein said means for pausing comprise means for pausing said transfer of data when said number of erasures exceeds said erasure threshold value.

13. (Original) The system of claim 9, wherein said means for determining is configured in hardware and a data path along which the data is transferred contains no processor.

14. (Original) The system of claim 9, wherein the data is transferred as part of a readcommand.

15. (Original) The system of claim 9, wherein the data is transferred as part of a verify command.

16. (Currently amended) A data-pausing system ~~computer program~~ stored on a computer-readable medium used in conjunction with a computing device, the ~~computer program~~ system comprising:

logic configured to establish a full error threshold value and an erasure threshold value;

logic configured to determine the number of full errors and the number of erasures in a codeword transferred from a data storage device to a host requesting one of a read command and a verify command;

logic configured to compare said number of full errors with said full error threshold value;

logic configured to compare said number of erasures with said erasure threshold value;

logic configured to pause the transfer of data if said number of full errors or erasures exceeds said full error threshold value or said erasure threshold value, respectively.

17. (Currently amended) The computer ~~program~~ system of claim 16, further comprising logic configured to interrogate registers which contain the data during the pausing of the transfer of data.

18. (Currently amended) A circuit for detecting errors in data and determining when to pause a transfer of the data and initiate an interrupt routine, the circuit comprising:

~~a host interface in communication with a host that outputs requests for data transfer procedures;~~

an error correcting code (ECC) encoder/decoder ~~in communication with said host interface, said ECC encoder/decoder~~ configured to detect and correct errors in a codeword being transferred;

a storage device interface in communication with said ECC encoder/decoder and a data storage device, said data storage device configured to store said codeword that is transferred to said host in response to said request; and

an interrupt initiation circuit in communication with said ECC encoder/decoder~~;~~, the interrupt initiation circuit comprising:

a processor;

an erasure threshold register configured to store a first threshold value;

a full error threshold register configured to store a second threshold value; and

a first comparator configured to compare said first threshold value with a number of erasures detected by said ECC encoder/decoder, said first comparator further configured to output an erasure indication that indicates whether or not said number of erasures exceeds said first threshold value;

~~a second comparator configured to compare said second threshold value with a number of full errors detected by said ECC encoder/decoder, said second comparator further configured to output a full error indication that indicates whether or not said number of full errors exceeds said second threshold value; and~~

~~logic circuitry configured to receive said erasure indication, said full error indication, and a status indication from said ECC encoder/decoder, said logic circuitry being further configured to process the received indications in response to a configuration request from a configuration register, said logic circuitry being further configured to initiate an interrupt of the processor that pauses the data transfer when the received indications meet criteria set by said configuration request~~.

19. (Original) The circuit of claim 18, wherein said first and second threshold values are established by said processor.

20. (Original) A method for determining when to pause the transfer of data, the method comprising:

establishing a full error threshold value;

establishing an erasure threshold value;

decoding a codeword being transferred from a data storage device;

indicating whether the codeword contains any errors;

determining the number of full errors;

determining whether the number of full errors exceeds the full error threshold value;

interrupting a processor and pausing the transfer of data if the number of full errors exceeds the full error threshold value;

determining the number of erasures;

determining whether the number of erasures exceeds the erasure threshold value;

interrupting the processor and pausing the transfer of data if the number of erasures exceeds the erasure threshold value.

21. (New) The circuit of claim 18, wherein the interrupt initiation circuit further comprises:

a second comparator configured to compare said second threshold value with a number of full errors detected by said ECC encoder/decoder, said second comparator further configured to output a full error indication that indicates whether or not said number of full errors exceeds said second threshold value; and

logic circuitry configured to receive said erasure indication, said full error indication, and a status indication from said ECC encoder/decoder, said logic circuitry being further configured to process the received indications in response to a configuration request from a configuration register, said logic circuitry being further configured to initiate an interrupt of the processor that pauses the data transfer when the received indications meet criteria set by said configuration request.